ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Proyecto Fin de Carrera

TÍTULO: Iphone Bookshelf

AUTOR: David Zamora Gutiérrez

DIRECTOR: Francesc Burrull i Mestres

Febrero / 2015

IPHONE APPLICATION

TOOLS

IPHONE

The iPhone is a line of Internet- and multimedia-enabled smartphones designed and marketed by Apple Inc. The first iPhone was unveiled by Apple CEO Steve Jobs on January 9, 2007, and released on June 29, 2007.

An iPhone can function as a video camera (video recording was not a standard feature until the iPhone 3GS was released), a camera phone, can send texts and receive visual voicemail, a portable media player, and an Internet client with email and web browsing capabilities, and both Wi-Fi and 3G connectivity. The user interface is built around the device's multi-touch screen, including a virtual keyboard rather than a physical one. Third-party as well as Apple application software is available from the App Store, which launched in mid-2008 and now has over 350,000 "apps" approved by Apple. These apps have diverse functionalities, including games, reference, GPS navigation, social networking, e-books...

To create applications for this device it's use the APPLE SDK.

APPLE SDK

The SDK basically consists of a set of tools that Apple provides to build, debug and test our developments. It contains the following programs:

- **XCODE**: Xcode is a suite of tools, developed by Apple, for developing software for Mac OS X and iOS.
- **Interface Builder**: Interface Builder is a software development application for Apple's Mac OS X operating system. It is part of Xcode (formerly Project Builder), the Apple Developer Connection developer's toolset. Interface Builder allows Cocoa and Carbon developers to create interfaces for applications using a graphical user interface. The resulting interface is stored as a .nib file, short for NeXT Interface Builder, or more recently, as a .xib file.
- The iphone emulator
- Instruments

OBJECTIVE C

Objective-C is a thin layer on top of C, and moreover is a strict superset of C; it is possible to compile any C program with an Objective-C compiler, and to freely include C code within an Objective-C class.

Objective-C derives its object syntax from Smalltalk. All of the syntax for non-object-oriented operations (including primitive variables, preprocessing, expressions, function declarations, and function calls) are identical to that of C, while the syntax for object-oriented features is an implementation of Smalltalk-style messaging.

The programmers used to use other programming languages like JAVA, C, or C++, should pay special attention to the following features:

Messages passing. The methods we are used to use in JAVA are called in Objective-C: messages, and the object or class which implement the method is called receiver. The syntax is:

```
[objectReceiver message];
```

The equivalent syntax in JAVA is using a point. So the next two sentences in Objective-C:

classReceiver obj= [objReceiver method1];
int integer = [obj method2];

In JAVA it would be:

int integer = objReceiver.method1( ).method2( );

In Objective_c is possible to send messages to nil (NULL) and the program doesn't show an error.

Methods. Methods in Objective-C use tags. All the methods can have tags in their parameters, which makes easier to understand the meaning of that parameter. For example:

Car *myCar = [[Car alloc] initWithXpos: 20 Ypos: 20 velocity:5.0
acceleration: 3.0];

The instance methods start with the sign '-'. The class methods start with the sign '+'.

Properties. Objective-C 2.0 introduces a new syntax to declare instance variables as properties, with optional attributes to configure the generation of accessor methods. Properties are, in a sense, public instance variables; that is, declaring an instance variable as a property provides external classes with access (possibly limited, e.g. read only) to that property.

- getter = nameMethod: To read dates from an object.
- setter = nameMethod: To modify dates.

By default, properties are considered atomic, which results in a lock preventing multiple threads from accessing them at the same time. A property can be declared as "nonatomic", which removes this lock.

Properties are implemented by way of the @synthesize keyword, which generates getter and setter methods according to the property declaration. Alternatively, the @dynamic keyword can be used to indicate that accessor methods will be provided by other means.

Properties can be accessed using the traditional message passing syntax, dot notation, or by name via the "valueForKey:"/"setValue:forKey:" methods.

In order to use dot notation to invoke property accessors within an instance method, the "self" keyword should be used.

A class or protocol's properties may be dynamically introspected*.

*In computing, type *introspection* is a capability of some object-oriented programming languages to determine the type of an object at runtime. This is a notable capability of the Objective-C language, and is a common feature in any language that allows object classes to be manipulated as first-class objects by the programmer.

Memory. The use of the memory is also different. In Objective-C is a programmer responsibility to erase the created objects and its variables. We use two methods to erase them:

- Using release, for example:
  nsstring *car;
  [ca release];
- By dealloc method. The programmer write the names of the variables to erase in that method, for example:
  - (void) dealloc {

```
[car release];
[super dealloc];
}
```

WEB SITE

TOOLS

CODEIGNITER

CodeIgniter is an open source web application framework for use in building dynamic web sites with PHP. "It's goal is to enable [developers] to develop projects much faster than writing code from scratch, by providing a rich set of libraries for commonly needed tasks, as well as a simple interface and logical structure to access these libraries."

The first public version of CodeIgniter was released on February 28, 2006. The latest stable version 2.0.2 was released April 07, 2011.

CodeIgniter is loosely based on the popular Model-View-Controller development pattern. While view and controller classes are a necessary part of development under CodeIgniter, Models are entirely optional and are rarely needed.

CodeIgniter is most often noted for its speed when compared to other PHP frameworks.

Model–view–controller (MVC) is a software architecture, currently considered an architectural pattern used in software engineering. The pattern isolates "domain logic" (the application logic for the user) from the user interface (input and presentation), permitting independent development, testing and maintenance of each (separation of concerns).
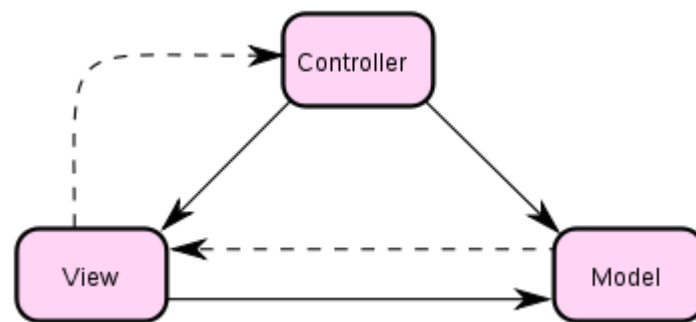
Model-view-controller concept. The solid line represents a direct association, the dashed an indirect association (via an observer for example)

The *model* manages the behavior and data of the application domain, responds to requests for information about its state (usually from the view), and responds to instructions to change state (usually from the controller). In event-driven systems, the model notifies observers (usually views) when the information changes so that they can react.

The *view* renders the model into a form suitable for interaction, typically a user interface element. Multiple views can exist for a single model for different purposes. A viewport typically has a one to one correspondence with a display surface and knows how to render to it.

The *controller* receives user input and initiates a response by making calls on model objects. A controller accepts input from the user and instructs the model and viewport to perform actions based on that input.

An MVC application may be a collection of model/view/controller triads, each responsible for a different UI element. The Swing GUI system, for example, models almost all interface components as individual MVC systems.

MVC is often seen in web applications where the view is the HTML or XHTML generated by the app. The controller receives GET or POST input and decides what to do with it, handing over to domain objects (i.e. the model) that contain the business rules and know how to carry out specific

tasks such as processing a new subscription, and which hand control to (X)HTML-generating components such as templating engines, XML pipelines, Ajax callbacks, etc.

The model is not necessarily merely a database; the 'model' in MVC is both the data and the business/domain logic needed to manipulate the data in the application. Many applications use a persistent storage mechanism such as a database to store data. MVC does not specifically mention the data access layer because it is understood to be underneath or encapsulated by the model. Models are not data access objects; however, in very simple apps that have little domain logic there is no real distinction to be made.

CodeIgniter has a fairly loose approach to MVC since Models are not required. If you don't need the added separation, or find that maintaining models requires more complexity than you want, you can ignore them and build your application minimally using Controllers and Views. CodeIgniter also enables you to incorporate your own existing scripts, or even develop core libraries for the system, enabling you to work in a way that makes the most sense to you.

<u>PHP</u>

PHP is a general-purpose scripting language originally designed for web development to produce dynamic web pages. For this purpose, PHP code is embedded into the HTML source document and interpreted by a web server with a PHP processor module, which generates the web page document. It also has evolved to include a command-line interface capability and can be used in standalone graphical applications. PHP can be deployed on most web servers and as a standalone interpreter, on almost every operating system and platform free of charge. A competitor to Microsoft's Active Server Pages (ASP) server-side script engine and similar languages, PHP is installed on more than 20 million websites and 1 million web servers.

PHP was originally created by Rasmus Lerdorf in 1995. The main implementation of PHP is now produced by The PHP Group and serves as the de facto standard for PHP as there is no formal specification. PHP is free software released under the PHP License; it is incompatible with the GNU General Public License (GPL) due to restrictions on the usage of the term PHP. While PHP originally stood for "Personal Home Page", it is now said to stand for "PHP: Hypertext Preprocessor", a recursive acronym.

Usage

PHP is a general-purpose scripting language that is especially suited to server-side web development where PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on web sites or elsewhere. It can also be used for command-line scripting and client-side GUI applications. PHP can be deployed on most web servers, many operating systems and platforms, and can be used with many relational database management systems (RDBMS). It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use.

PHP primarily acts as a filter, taking input from a file or stream containing text and/or PHP instructions and outputs another stream of data; most commonly the output will be HTML. Since

PHP 4, the PHP parser compiles input to produce bytecode for processing by the Zend Engine, giving improved performance over its interpreter predecessor.

Originally designed to create dynamic web pages, PHP now focuses mainly on server-side scripting, and it is similar to other server-side scripting languages that provide dynamic content from a web server to a client, such as Microsoft's Asp.net, Sun Microsystems' JavaServer Pages, and mod_perl. PHP has also attracted the development of many frameworks that provide building blocks and a design structure to promote rapid application development (RAD). Some of these include CakePHP, Symfony, *CodeIgniter*, and Zend Framework, offering features similar to other web application frameworks.

The LAMP architecture has become popular in the web industry as a way of deploying web applications. PHP is commonly used as the P in this bundle alongside Linux, *Apache* and *MySQL*, although the P may also refer to Python or Perl or some combination of the three. WAMP packages (Windows/ Apache/ MySQL / PHP), MAMP packages (Mac OS X / Apache / MySQL / PHP) and *XAMP* are also available.

As of April 2007, over 20 million Internet domains had web services hosted on servers with PHP installed and mod_php was recorded as the most popular Apache *HTTP* Server module. PHP is used as the server-side programming language on 75% of all web servers. Web content management systems written in PHP include MediaWiki, Joomla, eZ Publish, WordPress, Drupal and Moodle. All websites created using these tools are written in PHP, including the user-facing portion of Wikipedia, Facebook, and Digg.

Syntax

The PHP interpreter only executes PHP code within its delimiters. Anything outside its delimiters is not processed by PHP (although non-PHP text is still subject to control structures described within PHP code). The most common delimiters are <?php to open and ?> to close PHP sections. <script language="php"> and </script> delimiters are also available, as are the shortened forms <? or <?= (which is used to echo back a string or variable) and ?> as well as ASP-style short forms <% or <%= and %>. While short delimiters are used, they make script files less portable as support for them can be disabled in the PHP configuration, and so they are discouraged. The purpose of all these delimiters is to separate PHP code from non-PHP code, including HTML.

The first form of delimiters, <?php and ?>, in XHTML and other XML documents, creates correctly formed XML 'processing instructions'. This means that the resulting mixture of PHP code and other markup in the server-side file is itself well-formed XML.

Variables are prefixed with a dollar symbol and a type does not need to be specified in advance. Unlike function and class names, variable names are case sensitive. Both double-quoted ("") and heredoc strings allow the ability to embed a variable's value into the string. PHP treats newlines as whitespace in the manner of a free-form language (except when inside string quotes), and statements are terminated by a semicolon. PHP has three types of comment syntax: /* */ marks block and inline comments; // as well as # are used for one-line comments. The echo statement is one of several facilities PHP provides to output text.

In terms of keywords and language syntax, PHP is similar to most high level languages that follow the C style syntax. if conditions, for and while loops, and function returns are similar in syntax to languages such as C, C++, Java and Perl.

Data types

PHP stores whole numbers in a platform-dependent range, either a 64-bit or 32-bit signed integer equivalent to the C-language long type. Unsigned integers are converted to signed values in certain situations; this behavior is different from other programming languages. Integer variables can be assigned using decimal (positive and negative), octal, and hexadecimal notations. Floating point numbers are also stored in a platform-specific range. They can be specified using floating point notation, or two forms of scientific notation. PHP has a native Boolean type that is similar to the native Boolean types in Java and C++. Using the Boolean type conversion rules, non-zero values are interpreted as true and zero as false. The null data type represents a variable that has no value. The only value in the null data type is NULL. Variables of the "resource" type represent references to resources from external sources. These are typically created by functions from a particular extension, and can only be processed by functions from the same extension; examples include file, image, and database resources. Arrays can contain elements of any type that PHP can handle, including resources, objects, and even other arrays. Order is preserved in lists of values and in hashes with both keys and values, and the two can be intermingled. PHP also supports strings, which can be used with single quotes, double quotes, nowdoc or heredoc syntax.

The Standard PHP Library (SPL) attempts to solve standard problems and implements efficient data access interfaces and classes.

Functions

PHP has hundreds of base functions and thousands more via extensions. These functions are well documented on the PHP site (http://www.php.net/); however, the built-in library has a wide variety of naming conventions and inconsistencies. PHP currently has no functions for thread programming, although it does support multiprocess programming on POSIX systems. Additional functions can be defined by a developer:

```
function myFunction() {
return 'John Doe';
}
echo 'My name is ' . myFunction() . '!';
```

Objects

Basic object-oriented programming functionality was added in PHP 3 and improved in PHP 4. Object handling was completely rewritten for PHP 5, expanding the feature set and enhancing performance. In previous versions of PHP, objects were handled like value types. The drawback of this method was that the whole object was copied when a variable was assigned or passed as a parameter to a method. In the new approach, objects are referenced by handle, and not by value. PHP 5 introduced private and protected member variables and methods, along with abstract classes and final classes as well as abstract methods and final methods. It also introduced a standard way of declaring constructors and destructors, similar to that of other object-oriented languages such as C++, and a standard exception handling model. Furthermore, PHP 5 added interfaces and allowed for multiple interfaces to be implemented. There are special

interfaces that allow objects to interact with the runtime system. Objects implementing ArrayAccess can be used with array syntax and objects implementing Iterator or IteratorAggregate can be used with the foreach language construct. There is no virtual table feature in the engine, so static variables are bound with a name instead of a reference at compile time.

Visibility of properties and methods

The visibility of PHP properties and methods refers to visibility in PHP. It is defined using the keywords public, private, and protected. The default is public, if only var is used; var is a synonym for public. Items declared public can be accessed everywhere. Protected limits access to inherited classes (and to the class that defines the item). Private limits visibility only to the class that defines the item. Objects of the same type have access to each other's private and protected members even though they are not the same instance. PHP's member visibility features have sometimes been described as "highly useful". However, they have also sometimes been described as "at best irrelevant and at worst positively harmful."

Speed optimization

PHP source code is compiled on-the-fly to an internal format that can be executed by the PHP engine. In order to speed up execution time and not have to compile the PHP source code every time the webpage is accessed, PHP scripts can also be deployed in executable format using a PHP compiler.

Code optimizers aim to enhance the performance of the compiled code by reducing its size, merging redundant instructions and making other changes that can reduce the execution time. With PHP, there are often opportunities for code optimization. An example of a code optimizer is the eAccelerator PHP extension.

Another approach for reducing compilation overhead for PHP servers is using an opcode cache. Opcode caches work by caching the compiled form of a PHP script (opcodes) in shared memory to avoid the overhead of parsing and compiling the code every time the script runs. An opcode cache, APC, will be built into an upcoming release of PHP.

Opcode caching and code optimization can be combined for best efficiency, as the modifications do not depend on each other (they happen in distinct stages of the compilation).

Compilers

The PHP language was originally implemented using a PHP interpreter. Several compilers now exist, which decouple the PHP language from the interpreter. Advantages of compilation include not only better execution speed, but also obfuscation, static analysis, and improved interoperability with code written in other languages. The following are actual PHP compilers:

- Phc
- Roadsend
- Raven
- Phalanger
- Caucho Resin/Quercus
- HipHop

- php-to-scala   -   phc-win

"Embedders" also exist, which can package PHP scripts and other files into a Windows Executable, but do not provide any of the benefits of language-level compilation:
- Bambalam PHP EXE Compiler/Embedder
- ExeOutput for PHP

Resources

PHP includes free and open source libraries with the core build. PHP is a fundamentally Internet-aware system with modules built in for accessing FTP servers, many database servers, embedded SQL libraries such as embedded PostgreSQL, MySQL and SQLite, LDAP servers, and others. Many functions familiar to C programmers such as those in the stdio family are available in the standard PHP build.

PHP allows developers to write extensions in C to add functionality to the PHP language. These can then be compiled into PHP or loaded dynamically at runtime. Extensions have been written to add support for the Windows API, process management on Unix-like operating systems, multibyte strings (Unicode), cURL, and several popular compression formats. Some more unusual features include integration with Internet Relay Chat, dynamic generation of images and Adobe Flash content, and even speech synthesis. The PHP Extension Community Library (PECL) project is a repository for extensions to the PHP language.

HTTP

The Hypertext Transfer Protocol (HTTP) is a networking protocol for distributed, collaborative, hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web.

The standards development of HTTP has been coordinated by the Internet Engineering Task Force (IETF) and the World Wide Web Consortium (W3C), culminating in the publication of a series of Requests for Comments (RFCs), most notably RFC 2616 (June 1999), which defines HTTP/1.1, the version of HTTP in common use.

Technical overview

HTTP functions as a request-response protocol in the client-server computing model. In HTTP, a web browser, for example, acts as a client, while an application running on a computer hosting a web site functions as a server. The client submits an HTTP request message to the server. The server, which stores content, or provides resources, such as HTML files, or performs other functions on behalf of the client, returns a response message to the client. A response contains completion status information about the request and may contain any content requested by the client in its message body.

A client is often referred to as a user agent (UA). As well as web browsers, web crawlers are another common user agent. These include the indexing software used by search providers. Voice browsers are another less common but important class of user agent.

The HTTP protocol is designed to permit intermediate network elements to improve or enable communications between clients and servers. High-traffic websites often benefit from web cache servers that deliver content on behalf of the original, so-called origin server to improve response time. HTTP proxy servers at network boundaries facilitate communication when clients without a globally routable address are located in private networks by relaying the requests and responses between clients and servers.

HTTP is an Application Layer protocol designed within the framework of the Internet Protocol Suite. The protocol definitions presume a reliable Transport Layer protocol for host-to-host data transfer. The Transmission Control Protocol (TCP) is the dominant protocol in use for this purpose. However, HTTP has found application even with unreliable protocols, such as the User Datagram Protocol (UDP) in methods such as the Simple Service Discovery Protocol (SSDP).

HTTP Resources are identified and located on the network by Uniform Resource Identifiers (URIs)—or, more specifically, Uniform Resource Locators (URLs)—using the http or https URI schemes. URIs and the Hypertext Markup Language (HTML), form a system of inter-linked resources, called hypertext documents, on the Internet, that led to the establishment of the World Wide Web in 1990 by English physicist Tim Berners-Lee.

The original version of HTTP (HTTP/1.0) was revised in HTTP/1.1. HTTP/1.0 uses a separate connection to the same server for every request-response transaction, while HTTP/1.1 can reuse a connection multiple times, to download, for instance, images for a just delivered page. Hence HTTP/1.1 communications experience less latency as the establishment of TCP connections presents considerable overhead.

History

The term HyperText was coined by Ted Nelson who in turn was inspired by Vannevar Bush's microfilm-based "memex". Tim Berners-Lee first proposed the "WorldWideWeb" project — now known as the World Wide Web. Berners-Lee and his team are credited with inventing the original HTTP protocol along with the HTML and the associated technology for a web server and a text-based web browser. The first version of the protocol had only one method, namely GET, which would request a page from a server. The response from the server was always an HTML page.

The first documented version of HTTP was HTTP V0.9 (1991). Dave Raggett led the HTTP Working Group (HTTP WG) in 1995 and wanted to expand the protocol extended operations, extended negotiation, richer meta-information, tied with a security protocol and got more efficient by adding additional methods and header fields. RFC 1945 officially introduced and recognized HTTP V1.0 in 1996.

The HTTP WG planned to publish new standards in December 1995 and the support for pre-standard HTTP/1.1 based on the then developing RFC 2068 (called HTTP-NG) was rapidly adopted by the major browser developers in early 1996. By March 1996, pre-standard HTTP/1.1 was supported in Arena, Netscape 2.0, Netscape Navigator Gold 2.01, Mosaic 2.7, Lynx 2, and in Internet Explorer 3.0. End user adoption of the new browsers was rapid. In March 1996, one web hosting company reported that over 40% of browsers in use on the Internet were HTTP 1.1 compliant. That same web hosting company reported that by June 1996, 65% of all browsers

accessing their servers were HTTP/1.1 compliant. The HTTP/1.1 standard as defined in RFC 2068 was officially released in January 1997. Improvements and updates to the HTTP/1.1 standard were released under RFC 2616 in June 1999.

HTTP session

An HTTP session is a sequence of network request-response transactions. An HTTP client initiates a request. It establishes a Transmission Control Protocol (TCP) connection to a particular port on a host (typically port 80; see List of TCP and UDP port numbers). An HTTP server listening on that port waits for a client's request message. Upon receiving the request, the server sends back a status line, such as "HTTP/1.1 200 OK", and a message of its own, the body of which is perhaps the requested resource, an error message, or some other information.

HTML

HTML, which stands for HyperText Markup Language, is the predominant markup language for web pages. HTML is the basic building-blocks of webpages.

HTML is written in the form of HTML elements consisting of tags, enclosed in angle brackets (like <html>), within the web page content. HTML tags normally come in pairs like <h1> and </h1>. The first tag in a pair is the start tag, the second tag is the end tag (they are also called opening tags and closing tags). In between these tags web designers can add text, tables, images, etc.

The purpose of a web browser is to read HTML documents and compose them into visual or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts in languages such as JavaScript which affect the behavior of HTML webpages.

Web browsers can also refer to Cascading Style Sheets (CSS) to define the appearance and layout of text and other material. The W3C, maintainer of both the HTML and the CSS standards, encourages the use of CSS over explicitly presentational HTML markup.

History

In 1980, physicist Tim Berners-Lee, who was a contractor at CERN, proposed and prototyped ENQUIRE, a system for CERN researchers to use and share documents. In 1989, Berners-Lee wrote a memo proposing an Internet-based hypertext system. Berners-Lee specified HTML and wrote the browser and server software in the last part of 1990. In that year, Berners-Lee and CERN data systems engineer Robert Cailliau collaborated on a joint request for funding, but the project was not formally adopted by CERN. In his personal notes from 1990 he lists "some of the many areas in which hypertext is used" and puts an encyclopedia first.

The first publicly available description of HTML was a document called HTML Tags, first mentioned on the Internet by Berners-Lee in late 1991. It describes 20 elements comprising the

initial, relatively simple design of HTML. Except for the hyperlink tag, these were strongly influenced by SGMLguid, an in-house SGML based documentation format at CERN. Thirteen of these elements still exist in HTML 4.

HTML is a markup language that web browsers use to interpret and compose text, images and other material into visual or audible web pages. Default characteristics for every item of HTML markup are defined in the browser, and these characteristics can be altered or enhanced by the web page designer's additional use of CSS. Many of the text elements are found in the 1988 ISO technical report TR 9537 Techniques for using SGML, which in turn covers the features of early text formatting languages such as that used by the RUNOFF command developed in the early 1960s for the CTSS (Compatible Time-Sharing System) operating system: these formatting commands were derived from the commands used by typesetters to manually format documents. However, the SGML concept of generalized markup is based on elements (nested annotated ranges with attributes) rather than merely print effects, with also the separation of structure and processing; HTML has been progressively moved in this direction with CSS.

Berners-Lee considered HTML to be an application of SGML. It was formally defined as such by the Internet Engineering Task Force (IETF) with the mid-1993 publication of the first proposal for an HTML specification: "Hypertext Markup Language (HTML)" Internet-Draft by Berners-Lee and Dan Connolly, which included an SGML Document Type Definition to define the grammar. The draft expired after six months, but was notable for its acknowledgment of the NCSA Mosaic browser's custom tag for embedding in-line images, reflecting the IETF's philosophy of basing standards on successful prototypes. Similarly, Dave Raggett's competing Internet-Draft, "HTML+ (Hypertext Markup Format)", from late 1993, suggested standardizing already-implemented features like tables and fill-out forms.

After the HTML and HTML+ drafts expired in early 1994, the IETF created an HTML Working Group, which in 1995 completed "HTML 2.0", the first HTML specification intended to be treated as a standard against which future implementations should be based. Published as Request for Comments 1866, HTML 2.0 included ideas from the HTML and HTML+ drafts. The 2.0 designation was intended to distinguish the new edition from previous drafts.

Further development under the auspices of the IETF was stalled by competing interests. Since 1996, the HTML specifications have been maintained, with input from commercial software vendors, by the World Wide Web Consortium (W3C). However, in 2000, HTML also became an international standard (ISO/IEC 15445:2000). The last HTML specification published by the W3C is the HTML 4.01 Recommendation, published in late 1999. Its issues and errors were last acknowledged by errata published in 2001.

XHTML versions

- Main article: XHTML

XHTML is a separate language that began as a reformulation of HTML 4.01 using XML 1.0. It continues to be developed:
- XHTML 1.0, published January 26, 2000, as a W3C Recommendation, later revised and republished August 1, 2002. It offers the same three variations as HTML 4.0 and 4.01, reformulated in XML, with minor restrictions.
- XHTML 1.1, published May 31, 2001, as a W3C Recommendation. It is based on XHTML 1.0 Strict, but includes minor changes, can be customized, is reformulated using modules

from Modularization of XHTML, which was published April 10, 2001, as a W3C Recommendation.

- XHTML 2.0, There is no XHTML 2.0 standard. XHTML 2.0 is only a draft document and it is inappropriate to cite this document as other than work in progress. XHTML 2.0 is incompatible with XHTML 1.x and, therefore, would be more accurately characterized as an XHTML-inspired new language than an update to XHTML 1.x.
- XHTML5, which is an update to XHTML 1.x, is being defined alongside HTML5 in the HTML5 draft.

Elements

HTML documents are composed entirely of HTML elements that, in their most general form have three components: a pair of tags, a "start tag" and "end tag"; some attributes within the start tag; and finally, any textual and graphical content between the start and end tags, perhaps including other nested elements. The HTML element is everything between and including the start and end tags. Each tag is enclosed in angle brackets.

The general form of an HTML element is therefore: <tag attribute1="value1" attribute2="value2">content</tag>. Some HTML elements are defined as empty elements and take the form <tag attribute1="value1" attribute2="value2" />. Empty elements may enclose no content. The name of an HTML element is the name used in the tags. Note that the end tag's name is preceded by a slash character, "/", and that in empty elements the slash appears just before the closing >. If attributes are not mentioned, default values are used in each case.

Attributes

Most of the attributes of an element are name-value pairs, separated by "=" and written within the start tag of an element after the element's name. The value may be enclosed in single or double quotes, although values consisting of certain characters can be left unquoted in HTML (but not XHTML). Leaving attribute values unquoted is considered unsafe. In contrast with name-value pair attributes, there are some attributes that affect the element simply by their presence in the start tag of the element, like the ismap attribute for the img element.

There are several common attributes that may appear in many elements:

- The id attribute provides a document-wide unique identifier for an element. This is used to identify the element so that stylesheets can alter its presentational properties, and scripts may alter, animate or delete its contents or presentation. Appended to the URL of the page, it provides a globally unique identifier for the element, typically a sub-section of the page.
- The class attribute provides a way of classifying similar elements. This can be used for semantic or presentation purposes. For example, an HTML document might semantically use the designation class="notation" to indicate that all elements with this class value are subordinate to the main text of the document. In presentation, such elements might be gathered together and presented as footnotes on a page instead of appearing in the place where they occur in the HTML source. Class attributes are used semantically in microformats. Multiple class values may be specified; for example class="notation important" puts the element into both the 'notation' and the 'important' classes.

-   An author may use the style attribute to assign presentational properties to a particular element. It is considered better practice to use an element's id or class attributes to select the element from within a style sheet, though sometimes this can be too cumbersome for a simple, specific, or ad hoc styling.
-   The title attribute is used to attach sub textual explanation to an element. In most browsers this attribute is displayed as a tooltip.
-   The lang attribute identifies the natural language of the element's contents, which may be different from that of the rest of the document.

Character and entity references

As of version 4.0, HTML defines a set of 252 character entity references and a set of 1,114,050 numeric character references, both of which allow individual characters to be written via simple markup, rather than literally. A literal character and its markup counterpart are considered equivalent and are rendered identically.

The ability to "escape" characters in this way allows for the characters < and & (when written as < and &, respectively) to be interpreted as character data, rather than markup. For example, a literal < normally indicates the start of a tag, and & normally indicates the start of a character entity reference or numeric character reference; writing it as & or & or & allows & to be included in the content of an element or in the value of an attribute. The double-quote character ("), when used to quote an attribute value, must also be escaped as " or " or " when it appears within the attribute value itself. Equivalently, the single-quote character ('), when used to quote an attribute value, must also be escaped as ' or ' (not as ' except in XHTML documents) when it appears within the attribute value itself. If document authors overlook the need to escape such characters, some browsers can be very forgiving and try to use context to guess their intent. The result is still invalid markup, which makes the document less accessible to other browsers and to other user agents that may try to parse the document for search and indexing purposes for example.

Escaping also allows for characters that are not easily typed, or that are not available in the document's character encoding, to be represented within element and attribute content. For example, the acute-accented e (é), a character typically found only on Western European keyboards, can be written in any HTML document as the entity reference é or as the numeric references é or é, using characters that are available on all keyboards and are supported in all character encodings. Unicode character encodings such as UTF-8 are compatible with all modern browsers and allow direct access to almost all the characters of the world's writing systems.

Data types

HTML defines several data types for element content, such as script data and style sheet data, and a plethora of types for attribute values, including IDs, names, URIs, numbers, units of length, languages, media descriptors, colors, character encodings, dates and times, and so on. All of these data types are specializations of character data.

Document type declaration

HTML documents are required to start with a Document Type Declaration (informally, a "doctype"). In browsers, the doctype helps to define the rendering mode—particularly whether to use quirks mode.

The original purpose of the doctype was to enable parsing and validation of HTML documents by SGML tools based on the Document Type Definition (DTD). The DTD to which the DOCTYPE refers contains a machine-readable grammar specifying the permitted and prohibited content for a document conforming to such a DTD. Browsers, on the other hand, do not implement HTML as an application of SGML and by consequence do not read the DTD. HTML5 does not define a DTD, because of the technology's inherent limitations, so in HTML5 the doctype declaration, <!doctype html>, does not refer to a DTD.

Semantic HTML

Semantic HTML is a way of writing HTML that emphasizes the meaning of the encoded information over its presentation (look). HTML has included semantic markup from its inception, but has also included presentational markup such as <font>, <i> and <center> tags. There are also the semantically neutral span and div tags. Since the late 1990s when Cascading Style Sheets were beginning to work in most browsers, web authors have been encouraged to avoid the use of presentational HTML markup with a view to the separation of presentation and content.

In a 2001 discussion of the Semantic Web, Tim Berners-Lee and others gave examples of ways in which intelligent software 'agents' may one day automatically trawl the Web and find, filter and correlate previously unrelated, published facts for the benefit of human users. Such agents are not commonplace even now, but some of the ideas of Web 2.0, mashups and price comparison websites may be coming close. The main difference between these web application hybrids and Berners-Lee's semantic agents lies in the fact that the current aggregation and hybridisation of information is usually designed in by web developers, who already know the web locations and the API semantics of the specific data they wish to mash, compare and combine.

An important type of web agent that does trawl and read web pages automatically, without prior knowledge of what it might find, is the Web crawler or search-engine spider. These software agents are dependent on the semantic clarity of web pages they find as they use various techniques and algorithms to read and index millions of web pages a day and provide web users with search facilities without which the World Wide Web would be only a fraction of its current usefulness.

In order for search-engine spiders to be able to rate the significance of pieces of text they find in HTML documents, and also for those creating mashups and other hybrids as well as for more automated agents as they are developed, the semantic structures that exist in HTML need to be widely and uniformly applied to bring out the meaning of published text. Presentational markup tags are deprecated in current HTML and XHTML recommendations and are illegal in HTML5.

Good semantic HTML also improves the accessibility of web documents (see also Web Content Accessibility Guidelines). For example, when a screen reader or audio browser can correctly ascertain the structure of a document, it will not waste the visually impaired user's time by reading out repeated or irrelevant information when it has been marked up correctly.

MYSQL

MySQL is a database management system for data relational, multi-threaded and multi-user with more than six million installations. MySQL AB -from January 2008 a subsidiary of Sun Microsystems and a Oracle Corporation since April 2009- MySQL develops free software on a dual licensing scheme.

On one hand is available under the GNU GPL for any application compatible with this license, but for companies that want to incorporate it into products must purchase a special license that allows them to such use. It is developed mostly in ANSI.

Unlike projects like Apache, where the software is developed by a public community and copyrights are held code of the individual author, MySQL is sponsored by a private company, which owns the copyright to most of the code.

This is what enables the licensing scheme above. In addition to the sale of proprietary licenses, the company offers support and services. Hire workers for its operations around the world who collaborate via the Internet . MySQL AB was founded by David Axmark , Allan Larsson and Michael Widenius.

Project History

SQL (Structured Query Language) was first marketed in 1981 by IBM, which was submitted to ANSI and has since then been regarded as a standard for relational databases. Since 1986, the standard SQL has appeared in different versions such as: SQL: 92, SQL: 99, SQL: 2003. MySQL is an original idea from opensource company. MySQL AB initially established in Sweden in 1995 and whose founders are David Axmark, Allan Larsson and Michael "Monty" Widenius. The objective is that this company meets the standard MySQL SQL, but without sacrificing speed, reliability or usability.

Michael "Monty" Widenius in the 90's tried to use mSQL to connect the tables using low-level routines, ISAM, however, mSQL was not fast and flexible to your needs. This led him to create an SQL API called MySQL database very similar to mSQL but more portable.

MySQL is named after developer Michael Widenius' daughter, My. The SQL phrase stands for Structured Query Language.
The name of the MySQL Dolphin is Sakila and was selected by the founders of MySQL AB in the quiz "Name the Dolphin". This name was submitted by Ambrose Twebaze, a developer of open source software from Africa, derived from the language SiSwati, the local language of Swaziland and is the name of a town in Arusha, Tanzania, near the Uganda home town of Ambrose.

Programming languages

There are several APIs that allow applications written in various programming languages, access to MySQL databases, including C, C + +, C #, Pascal, Delphi (via dbExpress), Eiffel, Smalltalk, Java (with a native implementation of driver Java), Lisp, Perl, PHP, Python, Ruby, Gambas, REALbasic (Mac and Linux), FreeBASIC , and Tcl, each of these uses a specific API. There is also an interface ODBC, known MyODBC that allows any programming language that

supports ODBC to communicate with MySQL databases. You can also access the system from SAP, language ABAP .

Applications

MySQL is widely used in web applications such as Drupal or phpBB , platforms (Linux / Windows-Apache-MySQL-PHP / Perl / Python), and bug tracking tools like Bugzilla. Its popularity as a web application is closely related to PHP, which often appears in combination with MySQL. MySQL is a very quickly database in reading when the engine uses non-transactional MyISAM, but may cause integrity problems in environments with high competition of modifications. In Web applications there is low competition in the modification of data but there is intensive competition on reading, which makes MySQL ideal for such applications. Whatever environment you are using MySQL, it is important to advance on the performance monitoring to detect and correct errors of both SQL and programming.

Platforms

MySQL runs on multiple platforms, including:

- AIX
- BSD
- FreeBSD
- HP-UX
- GNU / Linux
- Mac OS X
- NetBSD
- Novell Netware
- OpenBSD
- OS / 2 Warp
- QNX

- SGI IRIX
- Solaris
- SunOS
- SCO OpenServer
- SCO UnixWare
- Tru64
- eBD
- Windows 95, 98, NT, 2000, XP, Vista, 7 and Server (2000 , 2003 and 2008)
- OpenVMS

Features

As of April 2009, MySQL offered MySQL 5.1 in two different variants: the open source MySQL Community Server and the commercial Enterprise Server. MySQL 5.5 is offered under the same licences. They have a common code base and include the following features:
- A broad subset of ANSI SQL 99, as well as extensions
- Cross-platform support
- Stored procedures
- Triggers
- Cursors
- Updatable Views
- True Varchar support
- Information schema

- Strict mode
- X/Open XA distributed transaction processing (DTP) support; two phase commit as part of this, using Oracle's InnoDB engine
- Independent storage engines (MyISAM for read speed, InnoDB for transactions and referential integrity, MySQL Archive for storing historical data in little space)
- Transactions with the InnoDB, BDB and Cluster storage engines; savepoints with InnoDB
- SSL support
- Query caching
- Sub-SELECTs (i.e. nested SELECTs)
- Replication support (i.e. Master-Master Replication & Master-Slave Replication) with one master per slave, many slaves per master, no automatic support for multiple masters per slave.
- Full-text indexing and searching using MyISAM engine
- Embedded database library
- Partial Unicode support (UTF-8 and UCS-2 encoded strings are limited to the BMP)
- Partial ACID compliance (full compliance only when using the non-default storage engines InnoDB, BDB and Cluster)
- Partititoned tables with pruning of partitions in optimiser
- Shared-nothing clustering through MySQL Cluster
- Hot backup (via mysqlhotcopy) under certain conditions.
- The developers release monthly versions of the MySQL Server. The sources can be obtained from MySQL's web site or from MySQL's Bazaar repository, both under the GPL license.

Distinguishing features

MySQL implements the following features, which some other RDBMS systems may not:
- Multiple storage engines, allowing one to choose the one that is most effective for each table in the application (in MySQL 5.0, storage engines must be compiled in; in MySQL 5.1, storage engines can be dynamically loaded at run time):
- Native storage engines (MyISAM, Falcon, Merge, Memory (heap), Federated, Archive, CSV, Blackhole, Cluster, Berkeley DB, EXAMPLE, Maria, and InnoDB, which was made the default as of 5.5)
- Partner-developed storage engines (solidDB, NitroEDB, Infobright (formerly Brighthouse), Kickfire, XtraDB, IBM DB2). InnoDB used to be a partner-developed storage engine, but with recent acquisitions, Oracle now owns both MySQL core and InnoDB.
- Community-developed storage engines (memcache engine, httpd, PBXT, Revision Engine)
- Custom storage engines
- Commit grouping, gathering multiple transactions from multiple connections together to increase the number of commits per second.

Support and licensing

MySQL offers support via their MySQL Enterprise product, including a 24/7 service with 30-minute response time. The support team has direct access to the developers as necessary to handle problems. In addition, it hosts forums and mailing lists, employees and other users are often available in several IRC channels providing assistance.

Buyers of MySQL Enterprise have access to binaries and software certified for their particular operating system, and access to monthly binary updates with the latest bug-fixes. Several levels of Enterprise membership are available, with varying response times and features ranging from how to and emergency support through server performance tuning and system architecture advice. The MySQL Network Monitoring and Advisory Service monitoring tool for database servers is available only to MySQL Enterprise customers.

Potential users can install MySQL Server as free software under the GNU General Public License (GPL), and the MySQL Enterprise subscriptions include a GPL version of the server, with a traditional proprietary version available on request at no additional cost for cases where the intended use is incompatible with the GPL.

Both the MySQL server software itself and the client libraries use dual-licensing distribution. Users may choose the GPL, which MySQL has extended with a FLOSS License Exception. It allows Software licensed under other OSI-compliant open source licenses, which are not compatible to the GPL, to link against the MySQL client libraries.

Customers that do not wish to follow the terms of the GPL may purchase a proprietary license.

Like many open-source programs, MySQL has trademarked its name, which others may use only with the trademark holder's permission.

History

In October 2005, Oracle Corporation acquired Innobase OY, the Finnish company that developed the third-party InnoDB storage engine that allows MySQL to provide such functionality as transactions and foreign keys. After the acquisition, an Oracle press release mentioned that the contracts that make the company's software available to MySQL AB would be due for renewal (and presumably renegotiation) some time in 2006. During the MySQL Users Conference in April 2006, MySQL issued a press release that confirmed that MySQL and Innobase OY agreed to a "multi-year" extension of their licensing agreement.

In February 2006, Oracle Corporation acquired Sleepycat Software, makers of the Berkeley DB, a database engine providing the basis for another MySQL storage engine. This had little effect, as Berkeley DB was not widely used, and was deprecated (due to lack of use) in MySQL 5.1.12, a pre-GA release of MySQL 5.1 released in October 2006.

In January 2008, Sun Microsystems bought MySQL for US$1 billion.

In April 2009, Oracle Corporation entered into an agreement to purchase Sun Microsystems, then owners of MySQL copyright and trademark. Sun's board of directors unanimously approved the deal, it was also approved by Sun's shareholders, and by the U.S. government on August 20, 2009. On December 14, 2009, Oracle pledged to continue to enhance MySQL as it had done for the previous four years. A movement against Oracle's acquisition of MySQL, to "Save MySQL" from Oracle was started by one of the MySQL founders, Monty Widenius. The petition of 50,000+ developers and users called upon the European Commission

to block approval of the acquisition. At the same time, several Free Software opinion leaders (including Eben Moglen, Pamela Jones of Groklaw, Jan Wildeboer and Carlo Piana, who also acted as co-counsel in the merger regulation procedure) advocated for the unconditional approval of the merger. As part of the negotiations with the European Commission, Oracle committed that MySQL server will continue to use the dual-licensing strategy long used by MySQL AB with commercial and GPL versions available until at least 2015. The Oracle acquisition was eventually unconditionally approved by the European Commission on January 21, 2010. Meanwhile, Monty Widenius has released a GPL only fork, MariaDB. MariaDB is based on the same code base as MySQL server and strives to maintain compatibility with Oracle provided versions.
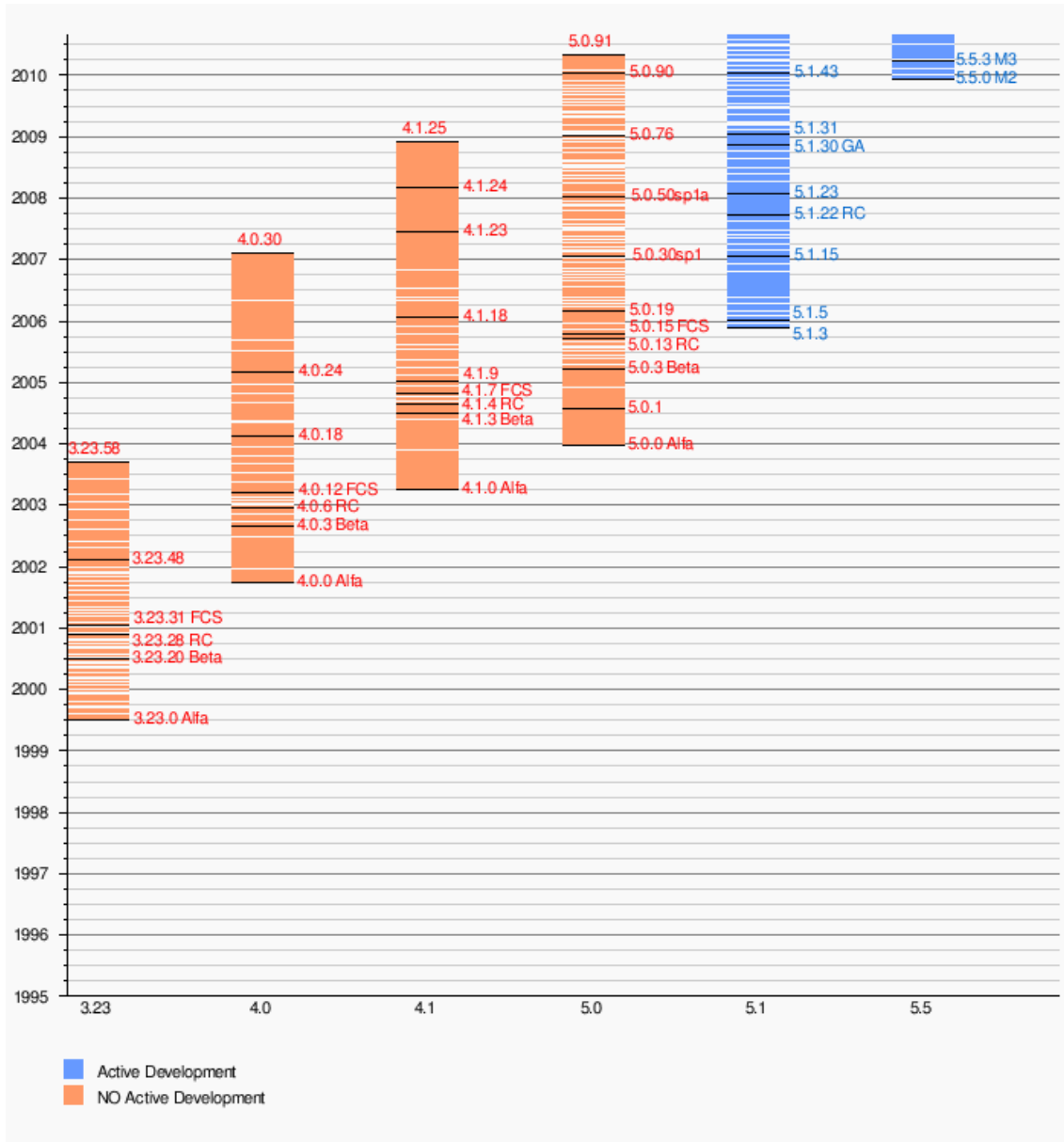
MSQL versions

APACHE

The Apache HTTP Server, commonly referred to as Apache, is web server software notable for playing a key role in the initial growth of the World Wide Web. In 2009 it became the first web server software to surpass the 100 million website milestone. Apache was the first viable alternative to the Netscape Communications Corporation web server (currently known as Oracle iPlanet Web Server), and has since evolved to rival other web servers in terms of functionality and performance. Typically Apache is run on a Unix-like operating system.

Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. The application is available for a wide variety of operating systems, including Unix, GNU, FreeBSD, Linux, Solaris, Novell NetWare, AmigaOS, Mac OS X, Microsoft Windows, OS/2, TPF, and eComStation. Released under the Apache License, Apache is characterized as open-source software.

Since April 1996 Apache has been the most popular HTTP server software in use. As of May 2011 Apache was estimated to serve 63% of all websites and 66% of the million busiest.

Features

Apache supports a variety of features, many implemented as compiled modules which extend the core functionality. These can range from server-side programming language support to authentication schemes. Some common language interfaces support Perl, Python, Tcl, and PHP. Popular authentication modules include mod_access, mod_auth, mod_digest, and mod_auth_digest, the successor to mod_digest. A sample of other features include SSL and TLS support (mod_ssl), a proxy module (mod_proxy), a URL rewriter (also known as a rewrite engine, implemented under mod_rewrite), custom log files (mod_log_config), and filtering support (mod_include and mod_ext_filter).

Popular compression methods on Apache include the external extension module, mod_gzip, implemented to help with reduction of the size (weight) of web pages served over HTTP. ModSecurity is an open source intrusion detection and prevention engine for web applications. Apache logs can be analyzed through a web browser using free scripts such as AWStats/W3Perl or Visitors.

Virtual hosting allows one Apache installation to serve many different actual websites. For example, one machine with one Apache installation could simultaneously serve www.example.com, www.test.com, test47.test-server.test.com, etc.

Apache features configurable error messages, DBMS-based authentication databases, and content negotiation. It is also supported by several graphical user interfaces (GUIs).

It supports password authentication and digital certificate authentication. Apache has a built in search engine and an HTML authorizing tool and supports FTP.

Usage

Apache is primarily used to send static and dynamic web pages on the World Wide Web. Many web applications are designed assuming as Apache deployment environment, or use features of this web server.

Apache is the web server component on the popular application platform LAMP, along with MySQL and programming languages PHP / Perl / Python (and now Ruby).

This web server is redistributed as part of various proprietary software packages, including database, Oracle and IBM WebSphere application server. Mac OS X integrates Apache as part of its own web server and in support of its application server, WebObjects. It is supported in some way by Borland in the Kylix development tools and Delphi. Apache is included with Novell NetWare 6.5, where is the default web server, and in many Linux distributions.

Apache is used for many other tasks where content needs to be made available in a safe and reliable. One example is the time to share files from a PC to the Internet. A user who has Apache installed on his desktop can place arbitrary files in the Apache document root, where they can be shared.

Web application programmers sometimes use a local version of Apache to preview and test code while it is developed.

Setting

Most of the configuration is made in the apache2.conf file or in httpd.conf file, depending on the system you are running. Any change in this file requires restarting the server, or forcing to read the configuration files again.

License

The software license under which the Apache Software Foundation is distributed is a distinctive part of the history of Apache HTTP Server and the open source community. The Apache license allows the distribution of derivative open and closed code from its original source.

The Free Software Foundation does not consider Apache License as compatible with version 2 of the GNU General Public License (GPL), in which the software licensed under the Apache License can't be integrated with software distributed under the GPL

However, the version 3 of the GPL includes a provision, which allows it to be compatible with licenses that have patent retaliation clauses, including the Apache License. The name Apache is a trademark and may be used only with permission from the owner of the mark.

CSS

Cascading Style Sheets (CSS) is a style sheet language used to describe the presentation semantics (the look and formatting) of a document written in a markup language. Its most common application is to style web pages written in HTML and XHTML, but the language can also be applied to any kind of XML document, including plain XML, SVG and XUL.

CSS is designed primarily to enable the separation of document content (written in HTML or a similar markup language) from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for tableless web design). CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. While the

author of a document typically links that document to a CSS style sheet, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified.

CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities or weights are calculated and assigned to rules, so that the results are predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998).

Syntax

CSS has a simple syntax and uses a number of English keywords to specify the names of various style properties. A style sheet consists of a list of rules. Each rule or rule-set consists of one or more selectors and a declaration block. A declaration-block consists of a list of declarations in braces. Each declaration itself consists of a property, a colon (:), a value. If there are multiple declarations in a block, a semi-colon (;) must be inserted to separate each declaration.

In CSS, selectors are used to declare which of the markup elements a style applies to, a kind of match expression. Selectors may apply to all elements of a specific type, or only those elements that match a certain attribute; elements may be matched depending on how they are placed relative to each other in the markup code, or on how they are nested within the Document Object Model.

Pseudo-classes are another form of specification used in CSS to identify markup elements, and in some cases, specific user actions to which a particular declaration block applies. A pseudo-class selects entire elements, such as :link or :visited, whereas a pseudo-element makes a selection that may consist of partial elements, such as :first-line or :first-letter. Selectors may be combined in other ways too, especially in CSS 2.1, to achieve greater specificity and flexibility.

Use

Prior to CSS, nearly all of the presentational attributes of HTML documents were contained within the HTML markup; all font colors, background styles, element alignments, borders and sizes had to be explicitly described, often repeatedly, within the HTML. CSS allows authors to move much of that information to a separate style sheet resulting in considerably simpler HTML markup.

Headings (h1 elements), sub-headings (h2), sub-sub-headings (h3), etc., are defined structurally using HTML. In print and on the screen, choice of font, size, color and emphasis for these elements is presentational.

In CSS, presentation is separated from structure. In print, CSS can define color, font, text alignment, size, borders, spacing, layout and many other typographic characteristics. It can do so

independently for on-screen and printed views. CSS also defines non-visual styles such as the speed and emphasis with which text is read out by aural text readers. The W3C now considers the advantages of CSS for defining all aspects of the presentation of HTML pages to be superior to other methods. It has therefore deprecated the use of all the original presentational HTML markup.

CSS files are inserted into HTML documents using the following syntax:

<link rel="stylesheet" href="http://example.com/css/style.css" type="text/css" />

History

Style sheets have existed in one form or another since the beginnings of SGML in the 1970s. Cascading Style Sheets were developed as a means for creating a consistent approach to providing style information for web documents.

As HTML grew, it came to encompass a wider variety of stylistic capabilities to meet the demands of web developers. This evolution gave the designer more control over site appearance but at the cost of HTML becoming more complex to write and maintain. Variations in web browser implementations made consistent site appearance difficult, and users had less control over how web content was displayed. Robert Cailliau wanted to separate the structure from the presentation. The ideal way would be to give the user different options and transferring three different kinds of style sheets: one for printing, one for the presentation on the screen and one for the editor feature.

To improve web presentation capabilities, nine different style sheet languages were proposed to the World Wide Web Consortium's list. Of the nine proposals, two were chosen as the foundation for what became CSS: Cascading HTML Style Sheets (CHSS) and Stream-based Style Sheet Proposal (SSP). CHSS, a language that has some resemblance to today's CSS, was proposed by Håkon Wium Lie in October 1994. Bert Bos was working on a browser called Argo, which used its own style sheet language called SSP. Lie and Yves Lafon joined Dave Raggett to expand the Arena browser for supporting CSS as a testbed application for the W3C. Lie and Bos worked together to develop the CSS standard.

Unlike existing style languages like DSSSL and FOSI, CSS allowed a document's style to be influenced by multiple style sheets. One style sheet could inherit or "cascade" from another, permitting a mixture of stylistic preferences controlled equally by the site designer and user.

Development of HTML, CSS, and the DOM had all been taking place in one group, the HTML Editorial Review Board (ERB). Early in 1997, the ERB was split into three working groups: HTML Working group, chaired by Dan Connolly of W3C; DOM Working group, chaired by Lauren Wood of SoftQuad; and CSS Working group, chaired by Chris Lilley of W3C.

The CSS Working Group began tackling issues that had not been addressed with CSS level 1, resulting in the creation of CSS level 2 on November 4, 1997. It was published as a W3C Recommendation on May 12, 1998. CSS level 3, which was started in 1998, is still under development as of 2009. In 2005 the CSS Working Groups decided to enforce the requirements

for standards more strictly. This meant that already published standards like CSS 2.1, CSS 3 Selectors and CSS 3 Text were pulled back from Candidate Recommendation to Working Draft level.

Variations

CSS has various levels and profiles. Each level of CSS builds upon the last, typically adding new features and typically denoted as CSS1, CSS2, and CSS3. Profiles are typically a subset of one or more levels of CSS built for a particular device or user interface. Currently there are profiles for mobile devices, printers, and television sets. Profiles should not be confused with media types, which were added in CSS2.

CSS 1

The first CSS specification to become an official W3C Recommendation is CSS level 1, published in December 1996. Among its capabilities are support for:

- Font properties such as typeface and emphasis
- Color of text, backgrounds, and other elements
- Text attributes such as spacing between words, letters, and lines of text
- Alignment of text, images, tables and other elements
- Margin, border, padding, and positioning for most elements
- Unique identification and generic classification of groups of attributes

The W3C no longer maintains the CSS1 Recommendation.

 CSS 2

CSS level 2 specification was developed by the W3C and published as a Recommendation in May 1998. A superset of CSS1, CSS2 includes a number of new capabilities like absolute, relative, and fixed positioning of elements and z-index, the concept of media types, support for aural style sheets and bidirectional text, and new font properties such as shadows. The W3C no longer maintains the CSS2 Recommendation.

CSS 3

Instead of defining all features in a single, large specification like CSS2, CSS3 is divided into several separate documents called "modules". Each module adds new capability or extends features defined in CSS2, over preserving backward compatibility. Work on CSS level 3 started around the time of publication of the original CSS2 Recommendation. The earliest CSS3 drafts were published in June 1999.

Due to the modularization, different modules have different stability and are in different status. As of March 2011, there are over 40 CSS modules published from the CSS Working Group. Some modules such as Selectors, Namespaces, Color and Media Queries are considered stable and are either in Candidate Recommendation or Proposed Recommendation status. Once CSS 2.1 is finalized and published as Recommendation, they are likely to go to Recommendation as well.

<u>Limitations</u>

- Poor controls for flexible layouts
- Selectors are unable to ascend
- Vertical control limitations
- Absence of expressions
- Lack of column declaration
- Cannot explicitly declare new scope independently of position
- Pseudo-class dynamic behavior not controllable

<u>Advantages</u>

- Flexibility
- Separation of content from presentation
- Site-wide consistency
- Bandwidth
- Page reformatting

<u>CSS frameworks</u>

CSS frameworks are pre-prepared libraries that are meant to allow for easier, more standards-compliant styling of web pages using the Cascading Style Sheets language. Layout-grid related CSS frameworks include Blueprint, 960 grid, and YUI CSS grids. Like programming and scripting language libraries, CSS frameworks are usually incorporated as external .css sheets referenced in the HTML <head>. They provide a number of ready-made options for designing and laying out the web page. While many of these frameworks have been published, some authors use them mostly for rapid prototyping, or for learning from, and prefer to 'handcraft' CSS that is appropriate to each published site without the design, maintenance and download overhead of having many unused features in the site's styling.

E-BOOK

An electronic book (also e-book, ebook, digital book) is a book-length publication in digital form, consisting of text, images, or both, and produced on, published through, and readable on computers or other electronic devices. Sometimes the equivalent of a conventional printed book, e-books can also be born digital. The Oxford Dictionary of English defines the e-book as "an electronic version of a printed book," but e-books can and do exist without any printed equivalent. E-books are usually read on dedicated hardware devices known as e-Readers or e-book devices. Personal computers and cell phones can be used to read e-books.

The potential for distance learning students to use e-books is explored. E-books are gaining wider interest since the introduction of portable electronic reading devices and software-based readers that provide users with more realistic book reading experiences. The paper discusses where to acquire e-book technology, and how to create e-books. It also reports an evaluation to test the usability of different types of e-book compiler software. By using one of the compilers, the use of e-books to improve the interaction between educators and distance learning

students in terms of access to teaching and learning materials and submission of assignments is also demonstrated.

What are e-books?

There is a growing interest in converting paper books to bytes as well as writing new titles in digital form. This in turn has resulted in a collection of hybrid definitions of e-books. Initially, paper books that had been converted to a digital format, usually through digitising processes which allow them to be displayed on computers, were defined as e-books. Then, the term also began to encompass multimedia, hypertext or hypermedia systems that are based on a book metaphor. Recently, the definition of an e-book has been extended to include book titles that are available online, that can be read as email, can be retrieved by a portable electronic reading device, or as a file that can be downloaded on to a computer. Another recent interpretation of an e-book is the "print-on-demand" book where the contents are stored in a system connected to a high-speed, high-quality printer, from which printed and bound copies are produced on demand with the possibility of buying chapter-by-chapter, customised books.

Formats

E-books are available in a wide range of formats, the simplest of which is plain ASCII-standard text. However, this format is extremely unappealing to read, cannot preserve formatting and cannot handle graphics. To solve these problems, the following formats can be used:

- Adobe Acrobat's Portable Document Format (PDF)
- Microsoft Reader's Literature (LIT)
- Rich Text Format (RTF)
- Night Kitchen's Tool Kit 3 (TK3)
- Markup Language (e.g. HyperText Markup Language - HTML, Standard Generalised Markup Language - SGML, eXtensible Markup Language - XML)
- Software for PDAs such as AportisDoc for Palm Pilots and Pocketbooks, Palm Reader and Mobi-Reader for Palm Handheld, Handspring Visor, and Window CE devices.

The most popular journal formats are either HTML or PDF.

Building E-Books with Compiler Software

Contents which have been developed and saved as HTML, text pages or sometimes PDF formats can be turned into a single executable file. This is achieved through the use of a program called an e-book compiler. A compiler can be defined as software that is designed to take a number of HTML or text pages and combine them into a single file. There are numerous compilers on the market:

- EBook Edit Pro
- Activ EBook
- HyperMaker HTML
- HTML Compiler Pro
- Illuminatus v4.51

- NeoBook
- Ebook Creator
- ebook Generator
- WinEBook

For students or individuals who cannot afford to buy commercial compilers, there are also free and trial compilers that can be downloaded from the Internet (e.g. Ebook Builder 4 and KeeBook Creator).

Microsoft has also developed its own compiler, ReaderWorks, which supports Microsoft Word, documents, text and HTML files, and the finished products are read using Microsoft Reader. Other tools that can be used to build e-books for Microsoft Reader include Microsoft Reader add-in for Microsoft Word 2000, Microsoft Reader Content SDK, and Mobipocket Publisher. To create PDF files, users can use Gymnast, freeware by Robert Schifreen, which supports hyperlinks, annotations, and automatic generation of bookmarks from headings.

Each compiler has its own set of features. Some have more features than others, some have similar features and some have unique features. Most e-book readers, for example, either hardware or software based, provide a personalised digital library where a collection of books is stored.

CODE

References

http://codeigniter.com/

http://www.desarrolloweb.com/articulos/modelo-vista-controlador-codeigniter.html

http://www.w3.org

http://www.faqs.org/rfcs/

http://httpd.apache.org/docs/2.0/mod/worker.html

http://journals.tdl.org/jodi/article/viewArticle/90/89

http://developer.apple.com/library/ios/#documentation/iPhone/Conceptual/iPhoneOSProgrammingGuide/Introduction/Introduction.html#//apple_ref/doc/uid/TP40007072

http://tutorialesiphone.wordpress.com/

http://es.wikipedia.org

http://en.wikipedia.org

http://php.net

http://mysql.com

http://w3schools.org

http://dessarrolloweb.com

http://www.xuletas.es

http://www.google.com

http://netbeans.org